Université d'Aix-Marseille III - Faculté des Sciences et Techniques de Saint-Jérôme

Licence Sciences et Technologies - Mention Maths-Info

I3 - Algorithmique et Programmation

TD n°4

1. Primitives de gestion de files

Dans ce problème, on s'intéresse à la gestion d'une file quand celle-ci est représentée sous forme de table. Le type C est celui donné en cours :

```
#define MAX ...
typedef ? element;  /* pour définir le type des éléments dans la file */
typedef struct {
                int debut, fin; /* indices des début et fin de file */
                int lg;  /* nombre d'éléments présents dans la file */
                element tab[MAX]; /* tableau contenant les éléments */
} File_tab;
```

Programmez les primitives spécifiées ci-dessous :

```
int successeur(int i)
/* spécifications : indice suivant de l'indice i */

int file_vide(File_tab f)
/* spécifications : évaluée à 1 si la file est vide et à 0 sinon */

int file_saturee(File_tab f)
/* spécifications : évaluée à 1 si la file est saturée et à 0 sinon */

void init_file(File_tab *f)
/* spécifications : initialise à vide la file pointée par f */

void enfile(File_tab *f, element e)
/* spécifications : ajoute l'élément e dans la file pointée par f */
/* sous réserve que la file ne soit pas saturée */

void defile(File_tab *f, element *e)
/* spécifications: enlève le premier élément de la file pointée par f : */
/* élément d'adresse e sous réserve que la file ne soit pas vide */
```

Réfléchissez aux traitements éventuellement nécessaires pour les cas limites pour enfile (saturation) et defile (file vide). Ne pourrait-on pas modifier certaines primitives (et leur entête) pour traiter ce type de problème ?

2. Fonctions et récursivité - Examen mai 2005 (6 pts)

Question 1. Définissez une fonction non-récursive prenant en entrée un entier strictement positif k et calculant la valeur de u_k où $u_0 = 1$ et $u_n = 2 * u_{n-1} * u_{n-1} + 1$ pour $n > 0$.

Question 2. Définissez cette fois-ci une fonction récursive prenant en entrée un entier strictement positif k et calculant la valeur de u_k où $u_0 = 1$ et $u_n = 2 * u_{n-1} * u_{n-1} + 1$ pour $n > 0$.

Question 3. Définissez une procédure récursive prenant en entrée un tableau d'entiers déclaré dans le type `tabint` défini ci-dessous et qui a pour résultat 1 si le tableau est trié dans l'ordre croissant, et 0 sinon.

```
#define n ...
typedef int tabint[n];
```

Vous pouvez avoir recours à plus d'un paramètre pour définir votre fonction. Vous donnerez également la forme d'un appel à cette fonction.